AD622077

TT-CS-64093

REPRINTED FROM

NAVAL RESEARCH LOGISTICS QUARTERLY

OFFICE OF NAVAL RESEARCH

NAVSO P-1278

VOL. 11, NOS. 2&3

JUNE-
SEPTEMBER 1964

HYPERBOLIC PROGRAMMING*

Bela Martos

Ministry of Metallurgy and Machine Industry and Computing Center of the Hungarian Academy of Sciences

Translated by

Andrew[†] and Veronika Whinston

*Graduate School of Industrial Administration
Carnegie Institute of Technology
Pittsburgh, Pennsylvania 15213*

INTRODUCTION

The Hyperbolic Programming Problem

In general, the role of mathematical programming can be summed up as follows: We are looking for a nonnegative solution of the variables $u_1, u_2, \ldots, u_n$ which satisfies the inequalities

$$
\begin{aligned}
f_1(u_1, u_2 \ldots u_n) &\leq 0 \\
\vdots \quad & \quad \vdots \\
f_m(u_1, u_2 \ldots u_n) &\leq 0,
\end{aligned}
$$

and which maximizes the given function $v = v(u_1, u_2 \ldots u_n)$.

A special case of this problem is known as linear programming, where the functions $f_1 \ldots f_m$ and v are linear. In this situation the given inequalities are

$$
\begin{aligned}
a_{11} u_1 + \ldots + a_{1n} u_n - a_{10} &\leq 0 \\
\vdots \qquad \qquad \vdots \qquad & \quad \vdots \\
a_{m1} u_1 + \ldots + a_{mn} u_n - a_{m0} &\leq 0,
\end{aligned}
$$

*This paper, which was written originally in Hungarian, represents a translation of the original document, published by the Hungarian Academy of Sciences in Publications of the Math. Inst. Hungarian Acad. Sci., 5, 383-406 (1960). The translators are grateful to Dr. Martos and to Professor W. W. Cooper for reviewing this translation and making numerous suggestions for its improvement.

This translation was undertaken as part of the contract "Planning and Control of Industrial Operations," with the Office of Naval Research at the Graduate School of Industrial Administration, Carnegie Institute of Technology, Management Sciences Research Group. Reproduction of this paper in whole or in part is permitted for any purpose of the United States Government. Contract Nonr 27 T024.

[†] University of Virginia.

which can be written as $u_1a_1 + u_2a_2 + \dots u_na_n \le a_o$, or in the following vector inequality form $Au \le a_o$. The function v can be written as $v = c_1u_1 + \dots + c_nu_n - c_o$ or $v = c^*u - c_o$.

For the solution of this problem, as it is known, Dantzig discovered the simplex method, which we presume to be known in what follows [2,3].

This study discusses another special case of mathematical programming, namely, when the functions $f_1 \dots f_m$ are linear and the function v is a linear fractional function. Since the graph of

$$y = \frac{cx - c_o}{dx - d_o},$$

which is a one-variable linear fractional function, is hyperbolic on the (x, y) coordinate system, we call this a hyperbolic programming problem.

We will show that the problem of hyperbolic programming can be solved with the aid of a slightly modified simplex method.

The method, to be explained, seems useful in the solution of economic problems where the different economical activities utilize fixed resources in proportion to the level of their values; the purpose of optimization, however, is not the definition of a revenue, allocation or economizing extremum (as it is with linear programming), but the extremum of a specific index number, and usually the most favorable ratio of revenues and allocations.

On the basis of this the problem of hyperbolic programming is the following: We are looking for the vector

$$u = \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ \cdot \\ u_n \end{bmatrix}$$

which maximizes the function

$$(1) \qquad v(u) = \frac{x(u)}{y(u)} = \frac{c^*u - c_o}{d^*u - d_o}$$

subject to the constraints $Au \le a_o$ and $u \ge 0$. The set of those points that satisfies the constraints we will call the set of feasible solutions and designate it as L. It is well known that L is a closed convex set with a finite number of extreme points.

Preliminary Observations

1. The necessary and sufficient conditions to reduce a hyperbolic programming problem to a linear programming problem could be easily defined. (I.e., the points of set R, to be defined later, should fall on the same line.) In practice, however, we do not think it necessary to exclude beforehand all degenerating cases, but we will exclude the following two trivial cases, (2) and (3):

The set L such that

(2) $$y = d^*u - d_o = \text{const.}$$

This is a linear programming problem. (If $y \equiv 0$, then the problem has no meaning.) The set L such that

(3) $$x(u) = v_o y(u),$$

where v_o = const. Thus on the entire set of L, $v(u) = v_o$ except for the common 0 points of $x(u)$ and $y(u)$ where, however, the function is not interpreted. In this case the programming problem has no meaning.

2. When the problem is not to find the maximum, but the minimum of the function v, we may equivalently consider the problem of maximization of $(-v)$.

3. We will assume, throughout, that there exists at least one point in the set L such that $y(u)$ is positive. This can be done without loss of generality since we may always consider the function $v = -x(u)/-y(u)$ instead of $v = x(u)/y(u)$. (We have already excluded the case $y \equiv 0$ in (2).)

Geometrical Interpretation

The problem can also be formulated as follows:

Let us consider the following transformation:

(4) $$w = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c^* \\ d^* \end{bmatrix} u - \begin{bmatrix} c_o \\ d_o \end{bmatrix}.$$

The transformation, as can be seen, consists of a linear transformation and a parallel shift. The transformation in (4) maps

$$L = \{u | Au \leq a_o, u \geq 0\}$$

a closed, convex set with finite extreme points to

$$R = \{w(u) | u \in L\}$$

along the set of the (y, x) plane. (As an exception we choose the y axis to be horizontal.) We can say the following about the set R:

1. It is closed and convex;
2. Each extreme point of the set R is an image of at least one extreme point of the set L;
3. It has a finite number of extreme points;
4. If L is convex polyhedron then R is a convex polygon; and
5. It has a point on the open half plane $y > 0$ (by "Preliminary Observations").

Statements 1 and 2 are derived directly from the known properties of linear transformations, and 3 and 4 from the previous statements.

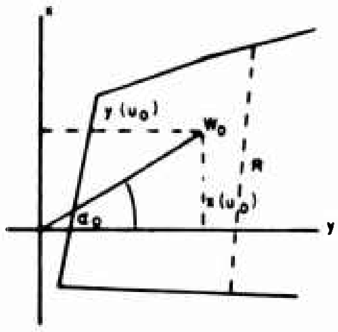

Figure 1

Let us consider the (y, x) plane. The value of the function $v(u_0) = x(u_0)/y(u_0)$ for some point u_0 in the set L can be found as the tangent of the angle α_0 which is constructed by drawing a half radius from the origin to the point w_0, which in turn is the image point of some point u_0. (See Figure 1. The dotted line indicates the case when R is bounded.) Thus the problem is to find such a point of the set R, the direction tangent of which is maximal, or to find such points of set L which were transformed to such maximal points of set R.*

THE SIMPLE CASE

The Conditions of the Simple Case

To simplify further discussion, we will present a simple case where the essence of the method to be applied may be observed.'

We will call the hyperbolic programming problem a simple case if it satisfies the following two conditions:

1. The set of possible solutions is bounded (therefore it is a convex polyhedral set, designated as P); and

2. The denominator will not be 0 on the set P. This constraint along with the assumption in the second section (Preliminary Observations) — considering the continuous function y — means that on the entire set P,

$$y = d^*u - d_0 > 0. \tag{5}$$

In the case of economical programming problem, the first condition means that none of the economical activities to be programmed can be unbounded. The assumption in the second condition excludes the possibility of the optimal value of the program becoming infinite. It can be seen that these conditions correspond to practical assumptions. Even for this reason alone the discussion of the simple problem has a practical interest.

Theorem Concerning the Maximum of Function v for the Simple Case

With the aid of the simplex method, we can examine the value of a function on the vertices of a convex polyhedral set. By the following theorem it is possible to apply the simplex method to the simple hyperbolic programming problem.

*There is another possible geometrical interpretation. Let us observe the family of hyperplanes which is defined by

$$(c_1 - \;\cdot d_1)u_1 + (c_2 - \;\cdot d_2)u_2 + \dots + (c_n - \;\cdot d_n)u_n \quad c_0 - \;\cdot d_0,$$

for different values of the parameter . We are looking for an element of the family of hyperplanes, at which the value of the parameter is both maximal and has a common point with the set L. Obviously, the optimal hyperplane contains an extreme point from the set L. The difference between the usual geometrical interpretation of linear programming and this case is that in the former the family of hyperplanes is parallel and in the latter it is not. This interpretation of the problem was worked out by Andras Prekopa. We will, however, employ the two dimensional transformation for the sake of simplicity in what follows.

'We consider it probable that the majority of the practical economical problems will satisfy those conditions which will be applied in this case.

THEOREM 1: If the linear fractional functional

$$v(u) = \frac{c^*u - c_o}{d^*u - d_o}$$

has the following properties:

1. the area of definition is a convex polyhedral P defined by an $Au \leq a_o$, $u \geq 0$ constraint; and

2. on the set P,

$$y = d^*u - d_o > 0,$$

then the function v has a finite maximum on the set P which is achieved on at least one vertex of the polyhedron.

PROOF OF THEOREM 1: Let $p_1, p_2 \ldots p_s$ be the vertices of the polyhedral P. Then, as it is known, any z of P can be written as a convex linear combination of these vectors,

$$z = \beta_1 p_1 + \beta_2 p_2 + \ldots + \beta_s p_s \tag{6}$$

where

$$\beta_1 + \beta_2 + \ldots + \beta_s = 1$$

and

$$\beta_i \geq 0, \quad i = 1, 2, \ldots s.$$

Let us assume that among the vertices, the value of the function v is the greatest at p_h, i.e.,

$$v(p_h) \geq v(p_i) \quad i = 1, 2, \ldots s, \tag{7}$$

that is

$$x(p_h)\, y(p_i) \geq y(p_h)\, x(p_i).$$

Multiply both sides by β_i and add from $i = 1$ to $i = s$

$$x(p_h) \Sigma_i \beta_i y(p_i) \geq y(p_h) \Sigma_i \beta_i\, x(p_i). \tag{8}$$

According to the following representation and by considering (6),

$$\Sigma_i \beta_i y(p_i) = \Sigma_i \beta_i \left(d^* p_i - d_o\right) = d^* \Sigma_i \beta_i p_i - d_o \Sigma_i \beta_i$$

$$= d^* z - d_o = y(z);$$

similarly,

$$\Sigma_i \beta_i\, x(p_i) = x(z).$$

Therefore, from (8)

$$x(p_h)\ y(z) \geq y(p_h)\ x(z),$$

from which follows

$$v(p_h) \geq v(z).$$

With this we proved our theorem and also the fact that with the aid of the simplex method there exists the possibility of solving a simple hyperbolic programming problem.

The Solution of a Simple Problem by the Simplex Method

We want to have an algorithm with the aid of which the number of necessary steps in the simplex method can be decreased and the cyclic iteration can be avoided.

In what follows, we assume that the inequalities have already been converted to equations by the known method of introducing slack variables. Then we can write the equations in this form $Au = a_o$. Let us assume that we know a basic solution of the problem containing m positive components. Let us further assume that exactly the first m components are positive. Then the basic vectors are $a_1, a_2, \ldots, a_m$.

For each a_i $(i = m + 1, m + 2, \ldots, n)$ let the coordinates in terms of the basic vectors be $b_{1i}, b_{2i}, \ldots, b_{mi}$ and for a_o the coordinates are $u_1, u_2, \ldots, u_m$.

Let

$$x_i = c_1 b_{1i} + \ldots + c_m b_{mi} - c_i,$$

$$y_i = d_1 b_{1i} + \ldots + d_m b_{mi} - d_i,$$

$$(i = m + 1, m + 2, \ldots, n), \text{ and}$$

$$t_i = xy_i - yx_i.$$

(If a_i is a basis-vector, then $b_{ji} = 0$, if $i \neq j$; $b_{ii} = 1$, and accordingly $x_i = y_i = t_i = 0$.)

Now we can write the following simplex tableau which is an extended one compared to the one used in linear programming:

c_i:			c_o	c_1	c_2	...	c_m	c_{m+1}	...	c_k	...	c_n
	d_i:		d_o	d_1	d_2	...	d_m	d_{m+1}	...	d_k	...	d_n
		basis	a_o	a_1	a_2	...	a_m	a_{m+1}	...	a_k	...	a_n
c_1	d_1	a_1	u_1	1	0	...	0	$b_{1,m+1}$	...	b_{1k}	...	b_{1n}
c_2	d_2	a_2	u_2	0	1	...	0	$b_{2,m+1}$	...	b_{2k}	...	b_{2n}
⋮	⋮	⋮	⋮	⋮	⋮		⋮	⋮		⋮		⋮
c_m	d_m	a_m	u_m	0	0	...	1	$b_{m,m+1}$	...	b_{mk}	...	b_{mn}
		x_i:	x	0	0	...	0	x_{m+1}	...	x_k	...	x_n
		x_i:	y	0	0	...	0	y_{m+1}	...	y_k	...	y_n
		t_i:	x/y	0	0	...	0	t_{m+1}	...	t_k	...	t_n

The arrangement of the known data is understood from the tableau. Into row t_i of column a_0 we write the function value $v = x/y$ given by the current basic solution.

Let us view the positive coordinates of the vector a_k and let

$$\min_{(b_{jk} > 0)} \frac{u_j}{b_{jk}} = \delta .$$

We have assumed that $u_j > 0$, therefore $\delta > 0$. (We will return to the question of degeneracy.)

As is known, the value of the linear functions $x(u)$ and $y(u)$ becomes

$$x' = x - \delta x_k$$

$$y' = y - \delta y_k$$

after the vector a_k has been brought into the basis.

Thus by exchanging the basis-vector the changed value of the function v is:

$$v' = \frac{x - \delta x_k}{y - \delta y_k} .$$

v' becomes greater than v, i.e., the changing to the new basis means approaching to the maximum if

$$\frac{x - \delta x_k}{y - \delta y_k} - \frac{x}{y} > 0,$$

or

$$\frac{\delta\left(xy_k - yx_k\right)}{y\left(y - \delta y_k\right)} > 0 .$$

Considering that in the case examined $y \geq 0$, $y - \delta y_k \geq 0$, and $\delta > 0$, we can say the following: if we bring into the basis a vector a_i having a positive coordinate for which the t_i is positive in the last row of the simplex tableau, $t_i = xy_i - yx_i > 0$, then the value of the function v will increase.

Therefore the simplex method employed in hyperbolic programming consists of bringing into the basis successively all positive t_i vectors (if they have positive coordinates). The method of calculating all u_j, b_{ji}, x, y, x_i, y_i values corresponds to the one used in linear programming; only $v = x/y$ and $t_i = xy_i - yx_i$ have to be computed by using these special formulas.

Since (1) the number of vertex points are finite, (2) the bringing in of each vector characterized by $t_i > 0$ increases the value of v, and (3) the function v will reach a maximum value at a certain vertex point, we must achieve an optimal solution after a finite number of steps by bringing the positive t_i vectors into the basis.

The tableau containing an optimal solution is an optimal tableau. From the above it follows that in the optimal tableau all $t_i \leq 0$. During the discussion of the general case (section entitled "Application of the Simplex Method to the Solution of the 'Good' Case of the General Problem," tableau 2) we will prove that this condition is sufficient: Each such tableau that contains $t_i \leq 0$ is an optimal tableau.

The number of computations necessary to solve a simple hyperbolic programming problem with the aid of the simplex method exceeds only by a small percentage the number of operations required for a linear programming problem of the same size. Programs for digital computers can be applied, with a slight alteration, to hyperbolic programming.

THE GENERAL PROBLEM

Definitions — Existence Theorems

In the case of the general problem we do not assume that the set L is bounded or that the denominator has no 0 point on the set L. What makes the discussion of the general problem necessary, as we will later see, is the fact that in certain cases where the conditions for the simple problem are absent the problem may still have an optimal solution even though the set L is unbounded or the denominator becomes 0.

Definitions.

1. We will call the "good" point of the set L where either $y(u) > 0$ or $y(u) = 0, x(u) < 0$. (According to the third observation under "Preliminary Observations," L always has a "good" point.)

2. We will call the "bad" point of the set L where either $y(u) < 0$ or $y(u) = 0, x(u) > 0$.

3. The singular point of the set L is at $y(u) = x(u) = 0$.

Therefore, in the transformation of the third section (Geometrical Interpretation), the good points are mapped on the inside of quadrants I and IV, on the positive side of the y axis (horizontal), and on the negative side of the x axis; the bad points are mapped on the inside of regions II and III, on the negative part of the y, and the positive part of the x axis; the singular points are mapped into the origin.

We will prove the following theorems:

THEOREM 2 ("BAD" CASE): If the set L has bad points then the function v is not bounded from above, and the programming problem has no optimal solution. (See Figures 2 and 3.)

THEOREM 3 ("GOOD" CASE): If the set L has only good points and the function v has a finite maximum on the set L, then this is taken on at least one extreme point of the set L. (See Figures 4 and 5.)

THEOREM 4 (SINGULAR CASE): If the set L has no bad points, but has singular points (and naturally good points) then it has a good point where the function v takes on a finite maximum value. (See Figures 6 and 7.)

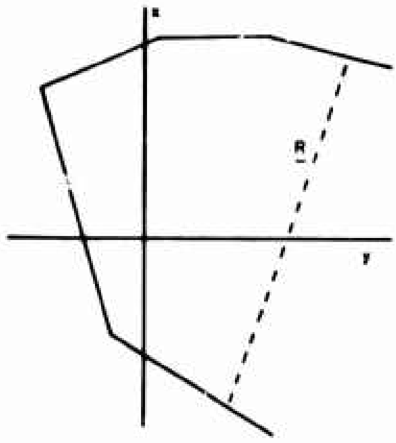

Figure 2

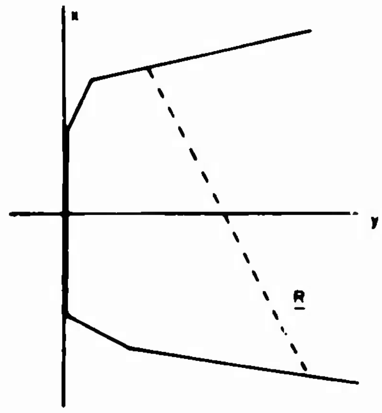

Figure 3

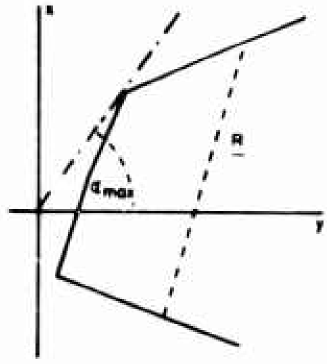

Figure 4

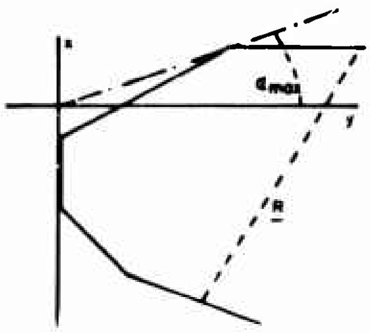

Figure 5

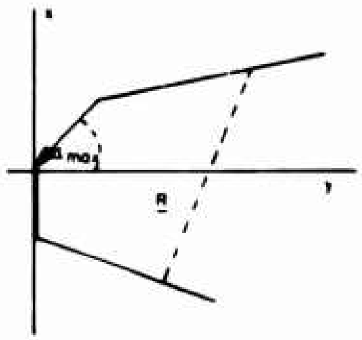

Figure 6

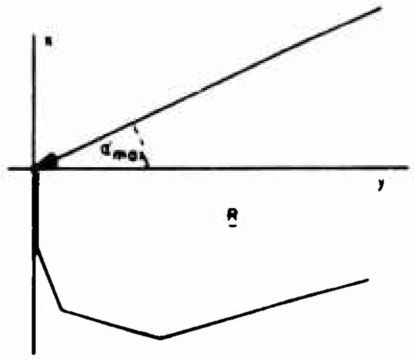

Figure 7

Proof of the Existence Theorems

PROOF OF THEOREM 2 (BAD CASE): On the basis of the statement supporting Eq. (3), for each bad point (P_r) of the set L we can choose such a good point (P_j) that on the line segment $\overline{P_rP_j}$ connecting these two points there will be no singular points. Therefore the line segment $\overline{P'_rP'_j}$ (the image on the plane of $\overline{P_rP_j}$) has a P' common point with the x axis which is different from the origin. If P' is on the positive x axis, then in the quadrant I, if it is on the negative x axis then in the quadrant III we can choose along the line segment $\overline{P'_rP'_j}$ such a series of points converging to P', along which series the direction tangent grows through positive numbers beyond all limits. Therefore the function v is not bounded from above, and the programming problem has no solution.

The theorem, of course, can be proved without the geometrical interpretation; however, we will exclude this method of proof.

PROOF OF THEOREM 3 (GOOD CASE): The theorem can be considered as the generalization of Theorem 1. Since if we do not declare the set L bounded, then the function v does not always have a maximum on the set L. Furthermore, as we will see, the function v, bounded from above, is not a sufficient condition for the existence of the maximum.

Those points of the set L for which $y = 0$, $x < 0$ cannot be optimal. These points we can approach only by such points the images of which are in quadrant IV. This way, however, the function v becomes infinite only through negative values, therefore it does not reach a maximum value. Consequently, it is sufficient to prove Theorem 3 for those points only for which

$$(9) \qquad y(u) = d^*u - d_o > 0 .$$

PROOF: Let $p_1, p_2, \ldots, p_s$ be the extreme points of the set L and the value of v be the highest at p_h, i.e.,

$$(10) \qquad v(p_h) \geq v(p_i), \qquad i = 1, 2, \ldots s .$$

We prove that if the set L has a point z where the function v is greater than at the chosen p_h extreme point, i.e.,

$$(11) \qquad v(z) > v(p_h),$$

then it also has such a point $\bar{z}$ where

$$(12) \qquad v(\bar{z}) > v(z).$$

Obviously z cannot be an extreme point of the set L. It follows that we can choose a p_g extreme point to point z, and a number $\alpha > 0$ so that if $q = z - p_g$, then $\bar{z} = z + \alpha q = (1+\alpha)z - \alpha p_g$ should still be a point of the set L; however,

$$(13) \qquad v(\bar{z}) = \frac{c^*\bar{z} - c_o}{d^*\bar{z} - d_o} = \frac{(1+\alpha)c^*z - \alpha c^*p_g - c_o}{(1+\alpha)d^*z - \alpha d^*p_g - d_o} = \frac{(1+\alpha)x(z) - \alpha x(p_g)}{(1+\alpha)y(z) - \alpha y(p_g)} .$$

From (11) and (10) it follows that $v(z) > v(p_g)$, i.e., $x(z)\,y(p_g) > x(p_g)\,y(z)$. When both sides are multiplied by $(-\alpha)$ and when $(1+\alpha)x(z)\,y(z)$ is added we obtain

$$x(z)\left[(1+\alpha)y(z) - \alpha y(p_g)\right] < y(z)\left[(1+\alpha)x(z) - \alpha x(p_g)\right] .$$

Equation (12) follows from this by considering (13). Thus the theorem is proved.

As far as the maximum of the function v is concerned the theorem permits three cases:

1. The function v reaches its maximum on the set L and it is reached on some extreme point of the set L (Figure 4).

2. The function v is not bounded on the set L from above (Figure 8).

3. The function v is bounded from above on the set L, but it does not reach its maximum (Figure 9).

These cases will be differentiated by examining the extreme points in a way to be given in the section entitled "Application of the Simplex Method to the Solution of the 'Good' Case of the General Problem."

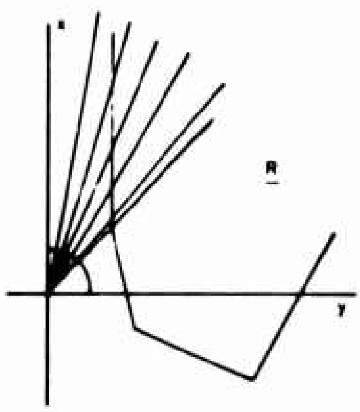

Figure 8

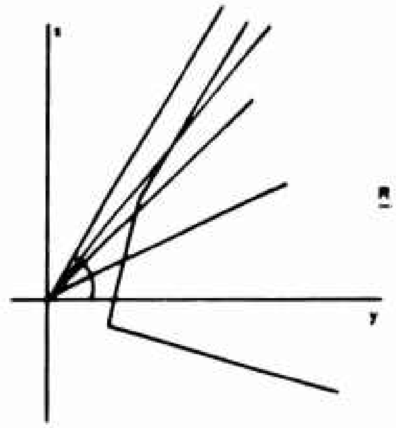

Figure 9

PROOF OF THEOREM 4 (SINGULAR CASE):

This theorem will be proved by the presentation of the optimal solution in the section entitled "Solution of the Singular Case."

<u>Preparation of the Solution: Differentiation of the Cases</u>

The hyperbolic programming problem solved by the simplex method can be initiated in the following steps:*

1. We define a starting solution by the same method that is used in linear programming. (The inequalities are converted to equations by the introduction of slack variables, and so on.)

2. We examine the y value belonging to the starting solution — (a) if y is positive we proceed to step 3; (b) if y is negative we proceed to functions $(-x)$ and $(-y)$ instead of functions x and y and continue with step 3; and (c) if y is 0 we see if the value of y could be increased by bringing any vector into the basis. If it can be increased we proceed according to (a); if it cannot be increased, only decreased, we act according to (b); if it cannot be either increased or decreased it corresponds to the excluded case $y \equiv 0$.

3. With linear programming we start searching for the minimum of the function y — (a) if in the meanwhile we reach a negative y value we interrupt the procedure, there is no solution (Bad case); (b) if $\min y > 0$ then proceed to step 5 (Good case); and (c) if $\min y = 0$ then proceed to step 4.

4. We add to the original given equations another condition,

$$d^{*}u = d_0 ,$$

and find, with linear programming, the maximum of the function x taking the added conditions into consideration' — (a) if, in the meanwhile, we reach a positive x value we interrupt the procedure, there is no solution (Bad case); (b) if $\max x < 0$, then proceed to step 5 (Good case); and (c) if $\max x = 0$, then proceed to the procedure to be described in the section entitled "Solution of the Singular Case."

*If we know from the character of the problem that $\min y > 0$ then these initiating steps are not needed.

'I.e., we are bringing those a_k vectors into the basis for which y_k 0, x_k 0.

5. In the course of the linear programming procedure, performed in step 3, we compute the corresponding $v = x/y$ value for each tableau. If 3(b) or 4(b) were the case, it would be best to choose as an initial tableau the one corresponding to the highest v value.

In case of a more complex problem, steps 3 and 4 should be performed by a computer.

Application of the Simplex Method to the Solution of the "Good" Case of the General Problem

It follows from Theorem 3 that in order to obtain an optimal point it is sufficient to examine the extreme points of the set L. Therefore the simplex method can be applied.

We remind the reader that in the section entitled "The Solution of a Simple Problem by the Simplex Method," we did not use the assumption that the set L is bounded and furthermore we needed only the milder assumption $y \geq 0$ instead of $y > 0$.

This assumption $(y \geq 0)$, however, holds for the "good" case of the general problem. Accordingly, the Simplex Method can be applied without alteration. What we do not know at this point is whether the problem has an optimal solution. Through the procedure, however, we will find the answer.

In the course of the procedure we may obtain the following tableaux:

T.1. In the last row of the tableau there is a $t_k > 0$ such that the corresponding a_k vectors have positive coordinates. In this case the given tableau is not optimal and the procedure is continued by bringing vector a_k into the basis.

T.2. There is no positive number in the last row of the tableau, all $t_i \leq 0$. In this case the function v reaches its maximum on the set L and the solution is optimal.

T.3. There is a $t_k > 0$ in the last row of the tableau, but one of the corresponding a_k vectors has no positive coordinate. In this case the function v does not reach the maximum on the set L, and the programming problem has no solution. We interrupt the procedure.

As can be seen, these three cases are identical with the ones in linear programming. The only difference is that in the case of T.3 the function v can be bounded from above, while the corresponding linear function cannot be. (See the discussion of the case T.3, below, and Figure 9.)

Proof for the T.1 Case. Our statement about the T.1 case does not need to be proved according to the section on the simplex method of solving simple problems. From the finite extreme points of the set L it follows that the T.1 type tableau may occur only in a finite number of steps; and afterwards we must arrive at a T.2 or T.3 type tableau.

Proof for the T.2 Case. Let us assume that

$$u^* = \left[u_1, u_2, \ldots, u_m, o, \ldots, o\right];$$

a possible solution appears in the tableau, i.e.,

$$\sum_{j=1}^{m} u_j a_j = a_o, \tag{14}$$

and there is no positive number in the last row of the tableau, i.e.,

$$t_i = x(u)\, y_i - y(u)\, x_i \leq 0 \quad (i = 1, 2, \ldots, n). \tag{15}$$

We prove that if $z^* = \left[z_1, z_2, \ldots, z_n\right]$ is a possible solution, i.e.,

$$\sum_{i=1}^{n} z_i a_i = a_o, \tag{16}$$

then

$$v(u) \geq v(z). \tag{17}$$

Any a_i vector on the $a_1, \ldots, a_m$ basis can be expressed as

$$a_i = \sum_{j=1}^{m} b_{ji} a_j, \qquad (i = 1, 2, \ldots, n);$$

inserted into (16) this gives

$$\sum_{i=1}^{n} z_i a_i = \sum_{i=1}^{n} z_i \sum_{j=1}^{m} b_{ji} a_j = \sum_{j=1}^{m} \left(\sum_{i=1}^{n} z_i b_{ji} \right) a_j = a_o.$$

If we combine this with (14) (and consider that a_o can be expressed only uniquely on the $a_1, \ldots, a_m$ basis) we get

$$\sum_{i=1}^{n} z_i b_{ji} = u_j. \tag{18}$$

By multiplying (15) by z_i and adding from $i = 1$ to $i = n$, we get

$$x(u) \sum_{i=1}^{n} y_i z_i \geq y(u) \sum_{i=1}^{n} x_i z_i. \tag{19}$$

Considering the definition of y_i and (18), we may transform

$$\left(\sum_{i=1}^{n} y_i z_i \right)$$

in the following manner:

(20) (Cont.)
$$\sum_{i=1}^{n} y_i z_i = \sum_{i=1}^{n} \left(\sum_{j=1}^{m} d_j b_{ji} - d_i \right) z_i = \sum_{j=1}^{m} d_j \left(\sum_{i=1}^{n} b_{ij} z_i \right) - \sum_{i=1}^{n} d_i z_i =$$

$$\text{(20)} \qquad = \left[\sum_{j=1}^{m} d_j u_j - d_o\right] - \left[\sum_{i=1}^{n} d_i z_i - d_o\right] = y(u) - y(z).$$

Similarly,

$$\text{(21)} \qquad \sum_{i=1}^{n} x_i z_i = x(u) - x(z).$$

By using (20) and (21) and substituting into (19), we get

$$x(u)\,[y(u) - y(z)] \leq y(u)\,[x(u) - x(z)],$$

that is,

$$\text{(22)} \qquad x(u)\,y(z) \geq y(u)\,x(z).$$

We consider that:

1. $y(u) = 0$ is impossible. Since, if $y(u) = 0$ then $x(u) < 0$ and choosing a negative $t_k = x(u)\,y_k - y(u)\,x_k = x(u)\,y_k < 0$ from which follows $y_k > 0$. This way, however, by bringing a_k into the basis we would reach a bad point, but the set does not have a bad point.

2. If $y(u) \neq 0$, $y(z) = 0$, then $v(z) = -\infty$ and thus (17) is true.

3. If $y(u) \neq 0$, and $y(z) \neq 0$, then (17) follows from (22).

With this our statement about the case T.2 has been proved. This proof also proves the penultimate paragraph of the section on the simplex method of solving simple problems.

Proof for the T.3 Case. To maintain the assumption in (14), let us assume that vector a_k $(k \geq m + 1)$ is the one for which

$$\text{(23)} \qquad t_k = x(u)\,y_k - y(u)\,x_k > 0,$$

and a_k, in terms of the given basis, has no positive coordinates, that is,

$$\text{(24)} \qquad a_k = \sum_{j=1}^{m} b_{jk}\,a_j,$$

where

$$\text{(25)} \qquad b_{jk} \leq 0, \qquad (j = 1, 2, \ldots, m).$$

Through multiplying Eq. (24) by a number $\lambda > 0$ and by adding Eq. (14) to it, we obtain

$$\sum_{j=1}^{m} \left(u_j - \lambda b_{jk}\right) a_j + \lambda a_k = a_o.$$

By considering (25), $u_j - \lambda b_{ji} > 0$, and thus the following,

$$z^* = \left[u_1 - \lambda b_{1k}, u_2 - \lambda b_{2k}, \ldots, u_m - \lambda b_{mk}, 0, \ldots, \lambda_k, \ldots, 0\right],$$

is a possible solution. The positive coordinates of this may take on any large value; therefore the set L is not bounded in this case.

Let us consider the function x evaluated at the point z:

$$x(z) = \sum_{j=1}^{m} c_j\left(u_j - \lambda b_{jk}\right) + \lambda c_k - c_o = \sum_{j=1}^{m} c_j u_j - c_o - \lambda\left[\sum_{j=1}^{m} c_j b_{jk} - c_k\right] = x(u) - \lambda x_k .$$

Similarly y(z) expressed, provides

$$v(z) = \frac{x(u) - \lambda x_k}{y(u) - \lambda y_k} . \tag{26}$$

The value of v(z) increases with the increase of λ, since considering (23)

$$\frac{\partial v}{\partial \lambda} = \frac{x(u)\,y_k - y(u)\,x_k}{\left[y(u) - \lambda y_k\right]^2} > 0 .$$

Since

$$y(z) = y(u) - \lambda y_k \geq 0 ,$$

inequality must hold at any $\lambda > 0$, it follows that

$$y_k \leq 0 :$$

1. If $y_k = 0$, then

$$v(z) = \frac{x(u) - \lambda x_k}{y(u)} ,$$

and according to (23) $\left[-y(u)\,x_k\right] > 0$, therefore $x_k < 0$. Thus with the increase of λ, v(z) increases without bound.

2. If $y_k < 0$, then, as can be easily shown,

$$v(z) < \frac{x_k}{y_k} = \lim_{\lambda \to +\infty} v(z) .$$

Therefore the function v is bounded from above on the set L, its lowest bound is x_k/y_k,* but it does not reach this value on any finite point of the set L. With this our statements for the T.3 case are proved.

*If a_k is the only vector satisfying the assumptions in T.3. If there are more such $a_{k1}, a_{k2}, a_{k\ell}$ vectors then the least upper bound is max x_{kf}/y_{kf} ($f = 1, 2, \ldots, \ell$).

Solution of the Singular Case

In the singular case the set L has singular points where the fractional function v is not interpreted, since both its numerator and denominator become 0. From a practical point of view this kind of problem should be considered solvable when there is such a point among the nonsingular points of the set L where the function v reaches its maximum value. We stated that if the set L has no bad points then there is always a solution in this sense. (In other words, even though the function has no boundary value in the singular point, it has lim. sup.)

To solve the singular case we start out from the tableau that was obtained in step 4(c) of "Preparation of the Solution: Differentiation of the Cases," in which $y = 0$ and $x = 0$. (Singular tableau.) From this it follows that in the last row of the singular tableau all $t_i = 0$ can be found. We have to return to the y_i and x_i values when examining the tableau:

1. The singular tableau, considering the minimum of the function y, is an optimal one, therefore each $y_i \leq 0$. (But among these there is $y_k < 0$, otherwise there were $y \equiv 0$.)

2. The singular tableau is an optimal one for the maximum problem of the function x which is enlarged by the assumption $d^*u = d_0$, i.e., for all such vectors to which $y_k = 0$ value belongs, $x_k \leq 0$.

There are two kinds of singular tableaux:

S.1. Some vector, having $y_k < 0$ and maximum x_k/y_k value has positive coordinates. Bringing this vector into the base we obtain an optimal solution immediately.

S.2. No vector, having $y_k < 0$ and maximum x_k/y_k value, has positive coordinates. As described below, we can again obtain an optimal (not basic) solution.

Proof for the S.1 Case. We proved in the T.2 case that a tableau is nonsingular and optimal if (1) $y > 0$ and (2) $t_i \geq 0$ $(i = 1, 2, \ldots, n)$.

1. After bringing in the a_k vector acceptable to the conditions in the S.1 case, the changed y' value of the new tableau will be positive

$$y' = y - \delta y_k = - \delta y_k > 0,$$

since $y_k < 0$. Therefore the new tableau is not singular.

2. We obtain the altered t_i' values as follows:

$$\left\{ \delta_i = \frac{b_{gi}}{b_{gk}} \text{, where } g \text{ is the index of the vector that was taken out of the basis.} \right\}$$

$$t_i' = x'y_i' - y'x_i' = - \delta x_k \left(y_i - \delta_i y_k\right) + \delta y_k \left(x_i - \delta_i x_k\right) = - \delta\left(x_k y_i - y_k x_i\right).$$

In order to prove that $t_i' \geq 0$ for all i, we have to show that

$$x_k y_i - y_k x_i \leq 0 \tag{27}$$

for all i, unless

$$\frac{x_k}{y_k} \leq \frac{x_i}{y_i}, \tag{28}$$

for all such i when the latter fraction is finite and not undetermined (i.e., if we brought in the vector having a maximal x_i/y_i).

Regarding the statement in (27),

1. is trivial in case $y_i = x_i = 0$,
2. case $y_i = 0$, $x_i < 0$ was excluded,
3. is true in case $y_i = 0$, $x_i > 0$ because $y_k < 0$, and
4. in case $y_i \neq 0$, x_i/y_i is finite and thus (27) follows from (28).

With this we proved that we reached an optimal tableau in the given manner.

<u>Proof for the S.2 Case</u>. Let us choose a vector a_k, ($y_k < 0$; $b_{jk} \leq 0$, $j = 1, 2, \ldots, m$) for which x_k/y_k value is maximal compared to all finite x_i/y_i values; i.e., (28) holds.

Thus, as we saw in the proof of the S.1 case,

$$x_k y_i \geq y_k x_i \qquad i = 1, 2, \ldots, n, \tag{27}$$

is true for all i. Let us assume that the basis of the singular tableau happens to be a_1, a_m and the solution in it is

$$s^* = \left[s_1, s_2, \ldots, s_m, 0, \ldots, 0\right].$$

Therefore,

$$x(s) = y(s) = 0.$$

Let us now formulate the solution

$$u^* = \left[s_1 - \lambda b_{1k}, s_2 - \lambda b_{2k}, \ldots, s_m - \lambda b_{mk}, 0 \ldots, \overset{(k)}{\lambda}, \ldots 0\right], \tag{29}$$

where λ is an arbitrary positive number. We showed in the proof of the T.3 case that the solution u is possible.

On the example of (26) we can see that

$$v(u) = \frac{x(s) - \lambda x_k}{y(s) - \lambda y_k} = \frac{x_k}{y_k}. \tag{30}$$

Now we will show that the solution u is optimal too. If

$$z^* = \left[z_1, z_2, \ldots, z_n\right]$$

is an arbitrary possible solution for which $y(z) < 0$, then

$$v(u) \geq v(z). \tag{31}$$

We multiply both sides of the inequality in (27) by z_i and add from $i = 1$ to $i = n$,

$$x_k \sum_{i=1}^{n} y_i z_i \geq y_k \sum_{i=1}^{n} x_i z_i. \tag{32}$$

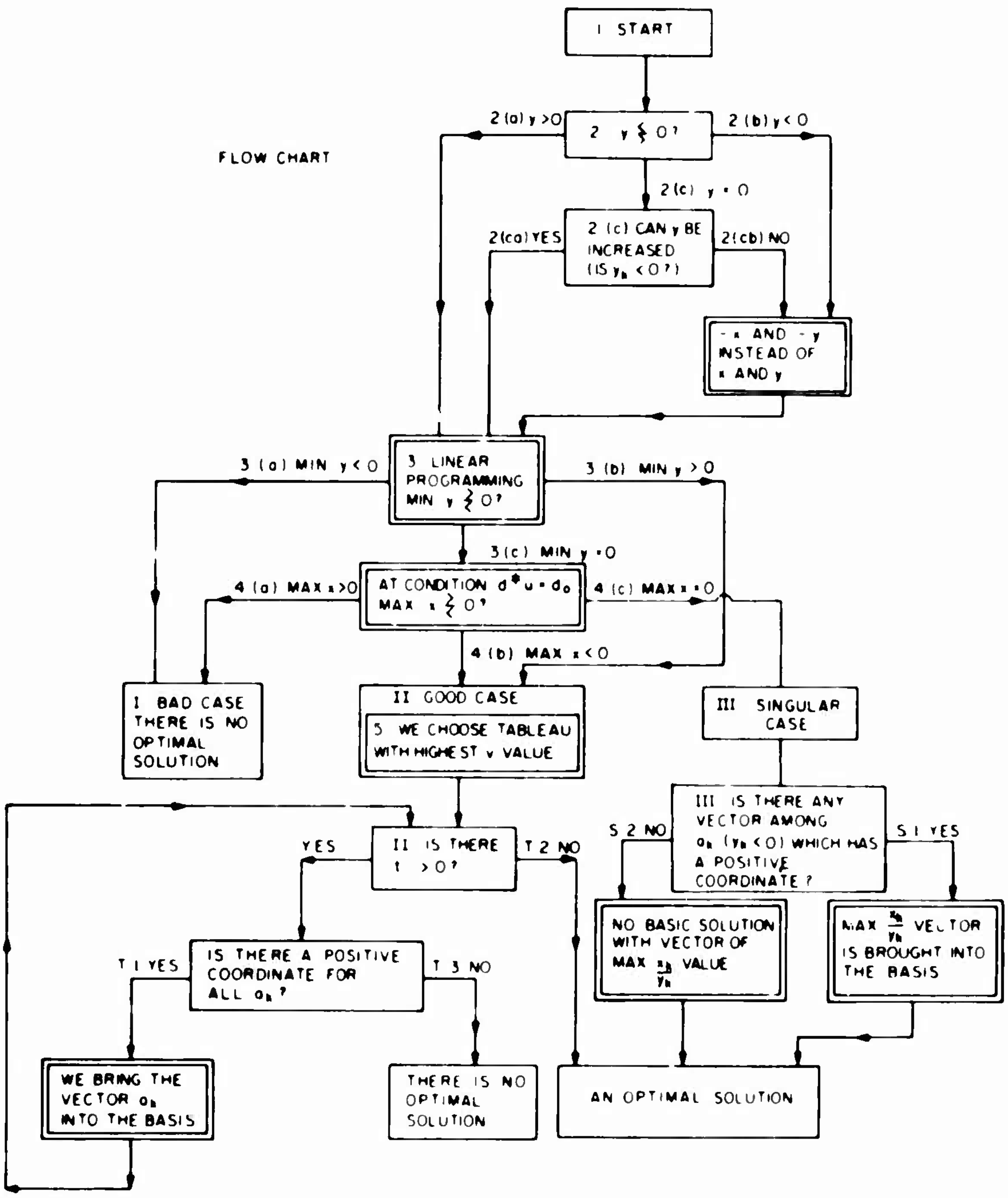
FLOW CHART
I START
2 y ≷ 0 ?
2 (a) y > 0
2 (b) y < 0
2 (c) y = 0
2 (c) CAN y BE INCREASED (IS y_h < 0 ?)
2 (ca) YES
2 (cb) NO
- x AND - y INSTEAD OF x AND y
3 LINEAR PROGRAMMING MIN y ≷ 0 ?
3 (a) MIN y < 0
3 (b) MIN y > 0
3 (c) MIN y = 0
AT CONDITION $d^*u = d_0$ MAX x ≷ 0 ?
4 (a) MAX x > 0
4 (b) MAX x < 0
4 (c) MAX x = 0
I BAD CASE THERE IS NO OPTIMAL SOLUTION
II GOOD CASE
5 WE CHOOSE TABLEAU WITH HIGHEST v VALUE
III SINGULAR CASE
III IS THERE ANY VECTOR AMONG a_h (y_h < 0) WHICH HAS A POSITIVE COORDINATE ?
S 2 NO
S 1 YES
NO BASIC SOLUTION WITH VECTOR OF MAX $\frac{x_h}{y_h}$ VALUE
MAX $\frac{x_h}{y_h}$ VECTOR IS BROUGHT INTO THE BASIS
II IS THERE t > 0 ?
YES
T 2 NO
IS THERE A POSITIVE COORDINATE FOR ALL a_h ?
T 1 YES
T 3 NO
WE BRING THE VECTOR a_h INTO THE BASIS
THERE IS NO OPTIMAL SOLUTION
AN OPTIMAL SOLUTION

Figure 13

$$\sum_{i=1}^{n} y_i z_i$$

can be changed, on the example of (20), to

$$\sum_{i=1}^{n} y_i z_i = y(s) - y(z) = - y(z).$$

Similarly,

$$\sum_{i=1}^{n} x_i z_i = - x(z).$$

Thus from (32) we have

$$- x_k\, y(z) \geq - y_k\, x(z).$$

Thus taking into consideration that $y_k < 0$ and $y(z) > 0$

$$\frac{x_k}{y_k} \geq \frac{x(z)}{y(z)}.$$

Therefore, considering (30), (31) is true and the solution of (29) is optimal.

We thus proved our statement for the S.2 case and at the same time produced the method for an optimal solution. With this Theorem 4, the existence theorem, was also proved.

Supplements

In this section we wish to give supplementary background, without proof, that is to say we wish to give solutions to a few problems that arise during hyperbolic programming.

The Case of Degeneracy. The degeneracy depends on the mutual position in space of the polyhedron generated by the column vectors of the matrix A and the a_0 vector. This position is the same in both hyperbolic and linear programming. It can be presumed that the degeneracy problems arising in the course of hyperbolic programming can usually be solved with the aid of the perturbation method worked out by Charnes [1]. We have to call attention, however, to the fact that in case of degeneracy the handling of singular tableaux becomes more complicated when the solution of the problem might be reached only by computing more successive singular tableaux. It remains to be examined whether in this case the perturbation method is appropriate for avoiding cycling.

The Enlarged Problem. In case not only

$$a_{i1} u_1 + a_{i2} u_2 + \dots + a_{in} u_n - a_{io} \leq 0$$

inequalities, but also

$$a_{i1} u_1 + a_{i2} u_2 + \dots + a_{in} u_n - a_{io} = 0$$

equations appear in the original conditions, we can change $Au \leq a_0$ into $Au + E\tau = a_0$. This way we obtain a starting solution easily and ensure the linear independence of the row vectors of the enlarged $[AE]$ matrix.

This way, however, the problem is interpreted on an enlarged set L' rather than on the original set L. If, originally, there were equations (and not inequalities) in rows $j_1, j_2, \ldots j_\ell$ then, in order to obtain optimal solution on the original L only, we would have to ensure that

$$\tau_{j1} = \tau_{j2} = \ldots = \tau_{j\ell} = 0$$

be in the optimal solution. For this reason the function to be maximized must be brought into the following form:

$$v = \frac{c_1u_1 + c_2u_2 - \ldots + c_nu_n - N\tau_{j1} - \ldots - N\tau_{j1} - c_o}{d_1u_1 + d_2u_2 + \ldots + d_nu_n - d_o},$$

where N is a very large positive number. As in linear programming there is no need for the numerical definition of N. The simplex tableau, however, must be enlarged by two rows which contain the coefficient of N in the expressions x_i and t_i.

All Optimal Solutions. In hyperbolic programming all optimal points form a convex set which has a finite number of extreme points. No extreme points belong to the set of optimal points in the S.2 singular case, since the optimum set is generated partially by infinitely distant points and partially by singular points.

The method for the definition of all optimal solutions is presented only for the case where the set of optimal points is bounded. If, in the procedure to be described, we reach an optimal tableau, having a vector for which $t_i = 0$ and all of its coordinates are not positive, then the set of optimal points is not bounded. This criterion holds even when we arrive at a singular tableau while determining all optimal points. (In the S.2 case, as we know, the optimal set is not bounded.)

If the above described case does not present itself then the set of all optimal points is bounded. In this case the extreme points of the set of optimal points can be obtained by starting from an optimal tableau and successively bringing into the basis all those vectors for which $t_i = 0$. In the meanwhile we may also obtain singular tableaux which should be taken into considerations in further operations. All optimal solutions can be obtained as convex linear combinations of solutions from all tableaux with the constraint that the coefficients of solutions from all nonsingular tableaux cannot be zero.

The image of the set of all optimal points may be:

1. an extreme point of the set R (Figure 10);
2. the closed line segment connecting two neighboring extreme points of the set R (Figure 11);
3. a half radius originating from one extreme point of the set R (Figure 12);
4. a half-open line segment connecting the origin with an extreme point of the set R (Figure 6); and
5. an open half radius starting from the origin (Figure 7).

Cases 3 and 5 can occur only when the set L is not bounded. Cases 4 and 5 can occur if, and only if, there is a singular one among the tableaux.

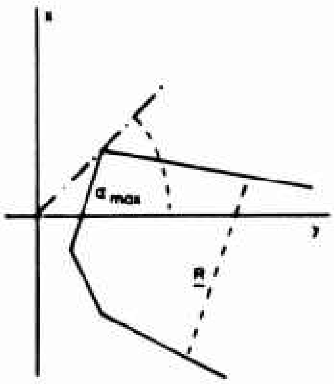

Figure 10

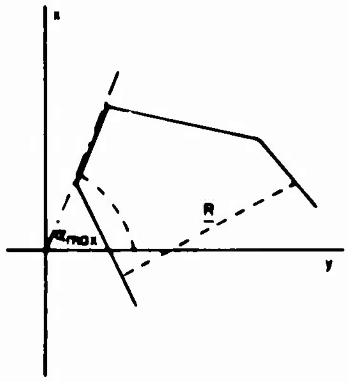

Figure 11

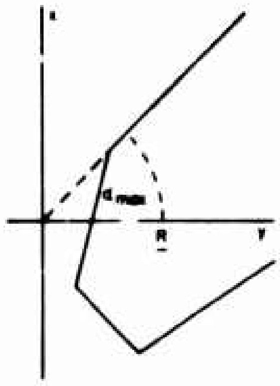

Figure 12

REFERENCES

[1] Charnes, A., Cooper, W. W., and Henderson, A., An Introduction to Linear Programming (John Wiley and Sons, N.Y., 1956).

[2] Koopmans, T. C., Activity Analysis of Production and Allocation (John Wiley and Sons, N.Y., 1959), Chap. XXI.

[3] Kreko, B. and Bacskay, Z., Introduction to Linear Programming (Economics and Law Publications, Budapest, 1957).

* * *